

In accordance with 37 C.F.R. §1.121, a claim listing including the status and text of all claims as currently presented appears below.

LISTING OF CURRENTLY PENDING CLAIMS

1-20. (Cancelled)

21. (Previously Presented) A method for providing exception handling for a computer program, said method comprising the steps of:
establishing a plurality of classes of exception types, said plurality of exception types including application exceptions, system exceptions and validation exceptions;
providing an exception dictionary used to list instances of each exception type;
throwing an exception;
identifying an exception type for the exception using the exception dictionary, wherein the identified exception type is one of an application exception, a system exception and a validation exception; and
providing an exception notice for the exception.

22. (Previously Presented) The method of claim 21, further comprising a step of capturing the exception.

23. (Previously Presented) The method of claim 21, further comprising the steps of:
determining in said identifying step that the exception is a validation exception;
and
continuing processing of a current module in the computer program.

24. (Previously Presented) The method of claim 21, further comprising the steps of:
determining in said identifying step that the exception is not a validation

exception; and
propagating the exception to a central place for handling.

25. (Previously Presented) The method of claim 21, further comprising a step of interfacing the exception handling features to a client/server system using an interface definition language.

26. (Previously Presented) The method of claim 25, wherein the interface definition language clears a set of operations, exceptions and attributes operating within the client/server system.

27. (Previously Presented) A system for providing exception handling for a computer program, comprising:

a class creator that establishes a plurality of classes of exception types, said plurality of exception types including application exceptions, system exceptions and validation exceptions;

an exception dictionary used to list instances of each exception type;

an exception determination mechanism that identifies an exception type for an exception using the exception dictionary, wherein the identified exception type is one of an application exception, a system exception and a validation exception; and

an exception notice generator that generates an exception notice for the exception.

28. (Previously Presented) The system of claim 27, further comprising an exception capture mechanism that captures an exception.

29. (Previously Presented) The system of claim 27, further comprising a processing mechanism that continues processing of a current module in the computer program when said exception determination mechanism identifies

the exception type to be a validation exception.

30. (Previously Presented) The system of claim 27, further comprising a processing mechanism that propagates the exception to a central place for handling when said exception determination mechanism identifies the exception type to be an exception other than a validation exception.
31. (Previously Presented) The system of claim 27, further comprising an interface definition language for specifying the interface of the exception handling system in a client/server system.
32. (Previously Presented) The system of claim 31, wherein said interface definition language clears a set of operations, exceptions and attributes operating within the client/server system.